II. AMENDMENTS TO THE CLAIMS

Please find below a complete listing of the claims in the application, including their status as effected by the present amendment:

1. (cancelled)

2. (cancelled)

3. (cancelled)

4. (cancelled)

5. (cancelled)

6. (currently amended) A method of executing a set of at least one incomplete task, comprising: ~~as defined in claim 1, wherein~~

(a) selecting an incomplete task from the set ~~includes selecting an incomplete task~~ on the basis of an expected duration for that task ; [[.]]

(b) resetting an execution timer having an expiry condition;

(c) advancing execution of the selected task until the earlier of (i) completion of the selected task or [[and]] (ii) expiry of the execution timer; and

(d) upon expiry of the execution timer prior to completion of the selected task, suspending execution of the selected task.

7. (cancelled)

8. (currently amended) A method of executing a set of at least one incomplete task, comprising: ~~as defined in claim 1, wherein~~

(a) selecting an incomplete task from the set ~~includes selecting an incomplete task~~ on the basis of a [[the]] number of times that the task has been previously suspended ; [[.]]

(b) resetting an execution timer having an expiry condition;

(c) advancing execution of the selected task until the earlier of (i) completion of the selected task or [[and]] (ii) expiry of the execution timer; and

(d) upon expiry of the execution timer prior to completion of the selected task, suspending execution of the selected task.

9. (cancelled)

10. (cancelled)

11. (currently amended) A method as defined in claim 6 [[1]], wherein advancing execution of the selected task includes beginning the selected task if the selected task has not been previously suspended.

12. (cancelled)

13. (original) A method as defined in claim 11, wherein advancing execution of the selected task includes resuming the selected task if the selected task has been previously suspended.

14. (original) A method as defined in claim 13, wherein suspending the selected task includes saving a context associated with the selected task.

15. (original) A method as defined in claim 14, wherein resuming the selected task includes retrieving the previously saved context associated with the selected task.

16. (original) A method as defined in claim 15, wherein the context associated with the selected task includes variables local to the selected task.

17. (original) A method as defined in claim 15, wherein the context associated with the selected task includes a state of the selected task.

18. (original) A method as defined in claim 15, wherein the context associated with the selected task includes a state of a central processing unit (CPU).

19. (currently amended) A method as defined in claim 6 [[1]], wherein the expiry condition of the execution timer is a pre-determined number of clock cycles.

20. (currently amended) A method as defined in claim 6 [[1]], wherein the expiry condition of the execution timer is a pre-determined period of time.

21. (currently amended) A method as defined in claim 6 [[1]], wherein the expiry condition of the execution timer is a pre-determined percentage of completeness of the selected task.

22. (currently amended) A method of executing a set of incomplete tasks, ~~as defined in claim 1, further~~ comprising:

(a) ~~if the selected task is a new version of an existing task in the set for which execution is more advanced than for the selected task,~~ removing an [[the]] existing incomplete task from the set when a newer version of the existing incomplete task is added to the set; [[.]]

(b) executing the remainder of the set of incomplete tasks.

23. (currently amended) A method as defined in claim 22 [[1]], wherein suspending the selected task includes saving a context associated with the selected task.

24. (original) A method as defined in claim 23, wherein the context associated with the selected task includes variables local to the selected task.

25. (original) A method as defined in claim 23, wherein the context associated with the selected task includes a state of the selected task.

26. (original) A method as defined in claim 23, wherein the context associated with the selected task includes a state of a central processing unit (CPU).

27. (cancelled)

28. (cancelled)

29. (cancelled)

30. (cancelled)

31. (cancelled)

32. (new) A method as defined in claim 8, wherein advancing execution of the selected task includes beginning the selected task if the selected task has not been previously suspended.

33. (new) A method as defined in claim 32, wherein advancing execution of the selected task includes resuming the selected task if the selected task has been previously suspended.

34. (new) A method as defined in claim 33, wherein suspending the selected task includes saving a context associated with the selected task.

35. (new) A method as defined in claim 34, wherein resuming the selected task includes retrieving the previously saved context associated with the selected task.

36. (new) A method as defined in claim 35, wherein the context associated with the selected task includes variables local to the selected task.

37. (new) A method as defined in claim 35, wherein the context associated with the selected task includes a state of the selected task.

38. (new) A method as defined in claim 35, wherein the context associated with the selected task includes a state of a central processing unit (CPU).

39. (new) A method as defined in claim 8, wherein the expiry condition of the execution timer is a pre-determined number of clock cycles.

40. (new) A method as defined in claim 8, wherein the expiry condition of the execution timer is a pre-determined period of time.

41. (new) A method as defined in claim 8, wherein the expiry condition of the execution timer is a pre-determined percentage of completeness of the selected task.